WHAT IS CLAIMED IS:

1. A method of supporting a kernel comprising:

generating a request in a kernel layer;

communicating the request to a user space;

processing the request in the user space to generate a response based on the request; and

communicating the response to the kernel layer.

2. The method of Claim 1, further comprising using the response in further processing in the kernel layer.

3. The method of Claim 1, further comprising;

generating the request at a kernel application driver; and

opening a communications channel between the kernel layer and user space at a bridge driver.

4. The method of Claim 3, wherein the kernel application driver and the bridge driver comprise portions of the same kernel.

5. The method of Claim 3, wherein the kernel application driver and bridge driver are separate kernels.

6. The method of Claim 3, further comprising queuing the request at the bridge driver.

7. The method of Claim 3, further comprising receiving the response from user space at the bridge driver in the kernel layer.

8. The method of Claim 3, further comprising:

receiving the request in the user space at a job manger; and

processing the request in the user space with a support library.

9. The method of Claim 8, further comprising queuing the request and the response in the user space.

10. The method of Claim 1, wherein said request is a request to verify an authentication parameter and wherein said response is whether the authentication parameter is verified.

11. The method of Claim 1, wherein said request is a request to generate an authentication parameter and wherein said response is the authentication parameter requested.

12. A system for extending kernel functionality comprising computer instructions stored on a computer readable storage medium and executable by a computer processor to:

generate a request in a kernel layer;

send the request to a user space;

process the request in the user space to generate a response; and

return the response to the kernel layer.

13. The system of Claim 12, wherein the computer instructions are further executable to open a communications channel between the kernel layer and the user space.

14. The system of Claim 12, wherein said computer instructions are further executable to queue said request and said response in the kernel layer.

15. The system of Claim 12, wherein said computer instructions are further executable to queue the request and the response in the user space.

16. The system of Claim 12, wherein said kernel layer comprises;

a kernel driver application; and

a bridge driver.

17. The system of Claim 16, wherein said kernel driver application is operable to generate the request.

18. The system of Claim 17, wherein said bridge driver is operable to:

establish a communications channel with the user space;

communicate the request to the user space; and

receive the response from the user space.

19. The system of Claim 18, wherein said bridge driver further comprises a kernel request queue and a kernel response queue and wherein said bridge driver is further operable to queue the request and the response in the kernel layer.

20. The system of Claim 17, wherein the user space further comprises:

a job manager operable to receive the request from the kernel layer; and

a support library operable to process the request and generate the response.

21. The system of Claim 20, wherein the user space further comprises a user space request queue and a user space response queue and wherein the job manager is further operable to queue the request and response in the user space.

22. The system of Claim 20, wherein said job manager is further operable to translate the request into a format usable by the support library.

23. The system of Claim 12, wherein the user space further comprises:

a job manager operable to receive the request from the kernel layer; and

a support library operable to process the request and generate the response.

24. The system of Claim 23, wherein the user space further comprises a user space request queue and a user space response queue and wherein the job manager is further operable to queue the request and response in the user space.

25. A method of authenticating a device over a network comprising:

generating a request in a kernel layer to perform a first portion of an authentication method in a user space;

communicating the request to the user space;

processing the request in the user space to generate a response;

communicating the response to the kernel layer; and

performing a second portion of the authentication method at the kernel layer.

26. The method of Claim 25, wherein the authentication method is one of kerberos, simple public-key, generic security service application programming interface (SPKM), Challenge Handshake Authentication Protocol (CHAP), or Secure Remote Password (SRP).

27. The method of Claim 25, further comprising:

receiving an authentication parameter from an initiator; and

wherein generating the request comprises a request to verify the authentication parameter in the user space.

28. The method of Claim 27, wherein the response comprises whether the first authentication parameter is verified.

29. The method of Claim 25, wherein:

the request comprises a request to generate an authentication parameter in the user space; and

the response includes the authentication parameter.

30. The method of Claim 29, further comprising communicating the authentication parameter to an initiator.

31. The method of Claim 25, further comprising generating the request at a kernel driver application.

32. The method of Claim 31, further comprising establishing a communications channel between the kernel layer and the user space.

33. The method of Claim 32, further comprising queuing the request and the response at the kernel layer.

34. A system of authenticating a device over a network comprising a set of computer instructions stored on a computer readable medium and executable by a computer processor to:

generate a request in a kernel layer to perform a first portion of an authentication method in a user space;

communicate the request to the user space;

process the request in user space to generate a response;

communicate the response to the kernel layer; and

perform a second portion of the authentication method at the kernel layer.

35. The system of Claim 34, wherein the computer instructions are further executable to receive an authentication parameter from an initiator.

36. The system of Claim 35, wherein:

the request is a request to verify the authentication parameter; and

the response comprises whether the authentication parameter is verified.

37. The system of Claim 33, wherein:

the request is a request to generate an authentication parameter in the user space; and

the response generated in the user space includes the authentication parameter.

38. The system of Claim 36, wherein the computer instructions are further executable to communicate the authentication parameter to an initiator.

39
~~38~~. The system of Claim 34, wherein the request is generated at a kernel driver application.

40
~~39~~. The system of Claim 34, wherein the computer instructions are further executable to establish a communications channel between the kernel layer and the user space.

Rule 126 cont

41
~~40~~. The method of Claim 34, wherein the computer instructions are further executable to queue the request and the response in the kernel layer.

42
41. A system of extending kernel functionality comprising:
comprising: /

a kernel driver application in a kernel layer operable to generate a request;

Rule 126 Cont

a bridge driver at the kernel layer operable to establish a communications channel between the kernel layer and a user space and communicate the request to the user space;

a support library in the user space operable to process the request in the user space and generate a corresponding response; and

a job manager in the user space operable to:

receive the request from the kernel layer;

forward the request to the support library; and

forward the response from the support library to the kernel layer.

43 42
42. The system of Claim 41, wherein the bridge driver is further operable to:

receive the response from the job manager; and

forward the response to the kernel driver application.

44 43
43. The system of Claim 42, wherein the bridge driver is further operable to queue the request and the response at the kernel layer.

45 44
44. The system of Claim 41, wherein the job manger is operable to queue the response and the request in the user space.

46 42
45. The system of Claim 41, wherein the job manger is operable to translate the request into a format usable by the

support library and the response into a format understandable to the bridge driver.

46 → 47. The system of Claim 41 → 42, wherein the request is a request to verify an authentication parameter and the response indicates whether the authentication parameter is verified.

Rule 126 cont

47 → 48. The system of Claim 41 → 42, wherein the request is a request to generate an authentication parameter and the response includes the authentication parameter.

48 → 49. The system of Claim 41 → 42, wherein the kernel driver application and the bridge driver are portions of the same kernel.

50. A system of authenticating a device over a network comprising:

an authentication engine in a kernel layer operable to:

generate a request to perform a first portion of an authentication method in a user space; and

perform a second portion of the authentication method at the kernel layer;

Rule 126 cont

a bridge driver at the kernel layer operable to establish a communications channel between the kernel layer and the user space and communicate the request to the user space;

a support library in the user space operable to process the request in the user space and generate a corresponding response; and

a job manager in the user space operable to:

receive the request from the kernel layer;

forward the request to the support library; and

forward the response from the support library to the kernel layer.

51. The system of Claim 50, wherein the bridge driver is further operable to:

receive the response from the job manager; and

forward the response to the authentication engine.

52. The system of Claim 51, wherein the bridge manager is further operable to queue the request and the response at the kernel layer.

53. The system of Claim 52, wherein the job manager is further operable to queue the request and the response in the user space.

53. The system of Claim 49 further comprising:

a network interface at the kernel layer operable to receive an authentication request from an initiator and forward authentication parameters to the authentication engine.

54. The system of Claim 53 further comprising:

a configuration database accessible by the authentication engine, a set of initiator names corresponding to initiators that can be authenticated and a set of allowable authentication methods.

Rule 126 cont

56. The system of Claim 54, wherein the authentication parameters include an initiator name and wherein the authentication engine is operable to compare the received initiator name to the set of initiator names in the configuration database.

56. The system of Claim 54, wherein the authentication engine if further operable to negotiate the authentication method with the initiator based on the set of allowable authentication methods.

57. The system of Claim 56, wherein the authentication method is secure remote password.

58. The system of Claim 53, wherein:

the authentication engine is further operable to generate a request to verify at least one of the authentication parameters;

the support library is operable verify the at least one of the authentication parameters.

60 59. The system of Claim 58, the support library is operable to verify a public ephemeral key.

61 50. The system of Claim 49, wherein the first portion of the authentication method includes the generation of an authentication parameter.

Rule 26

62 61. The system of Claim 60, wherein the support library is operable to generate the authentication parameter.

63 62. The system of Claim 61, wherein the response includes the authentication parameter.

64 63. The method of Claim 62, wherein the authentication engine is further operable to hash the authentication parameter.

65 63. The method of Claim 62, wherein the authentication engine is further operable to return the authentication parameter to an initiator.